REMARKS/ARGUMENTS

Claims 1-45 are now pending, a total of 45 claims. Claims 1 and 24 are independent. Claims 38-40 are indicated as reciting allowable subject matter.

Applicant respectfully requests reconsideration of the application. For reasons discussed in the accompanying Petition to Redate Office Action, this paper is timely.

I. Status of Previous Papers

Formal drawings, a Preliminary Amendment, and an Information Disclosure Statement and Forms 1449 were submitted October 31, 2000. Initialed copies of these Forms 1449 have been received, and thus this mailing was received by the PTO. However, the Preliminary Amendment and Formal Drawings have not been acknowledged.

- For the Examiner's convenience and ready reference, a replacement copy of the January 2001 Preliminary Amendment is enclosed herewith. I certify that this Preliminary Amendment was deposited with the United States Postal Service on October 31, 2000 as First Class Mail in an envelope with sufficient postage addressed to The Commissioner for Patents, Washington D.C. 20231, and is entitled to entry as of that date, pursuant to 37 C.F.R. § 1.8.

- Applicant requests confirmation that this Preliminary Amendment is entered.

- Applicant requests confirmation that formal drawings were received in or around November 2000, and entered.

Similarly, an IDS and Form 1149 were filed on April 30, 2003. A checked-off copy of this form 1449 has not been received. The references from that Form 1449 are resubmitted on the Form 1449 submitted herewith.

The FAX of the Office Action of January 28, 2004 included none of the ancillary papers (mailing page, any form PTO-892, etc.). Applicant renews the request of December 8, 2003, for a remailing of the complete Office Action, with all ancillary papers, so that Applicant's records are complete.

II. Paragraph 2 of the Office Action

Paragraph 2 of the Office Action invokes 35 U.S.C. § 112, second paragraph. MPEP § 2174 notes that the requirement for clarity of the claims under § 112 <u>second</u> paragraph is

distinct from the requirements for specification support under § 112 <u>first</u> paragraph. In contrast,

the Office Action discusses only the <u>specification</u>. The Office Action does not identify any

difficulty that the public would have in determining the scope of any <u>claim</u> language. Without

an identification of an ambiguity, Applicant is unable to show how the claims overcome that

ambiguity. Applicant submits that no rejection exists.

In order to advance prosecution, Applicant notes that hardware control of profiling is

taught at Figs. 4g, 4h, 4i, 5a and 5b, and the associated portions of the specification, at pages 73-

89, among other portions.

III.   Paragraph 3 of the Office Action

Paragraph 3 of the Office Action raises a question under § 103(a) based on the O'Donnell

'369 patent and Conte's article "Using Branch Handling Hardware to Support Profile-Driven

Optimization." Applicant traverses on two separate grounds.

Claim 1 recites as follows:

1.   A method, comprising:

while executing a program on a computer, detecting the occurrence of
profileable events occurring in the instruction pipeline, and directing the
instruction pipeline to record profile information describing the profileable events
essentially concurrently with the occurrence of the profileable events, the
detecting and recording occurring under control of hardware of the computer
without compiler assistance for execution profiling or event-by-event software
intervention.

A.   Conte does not provide motivation to modify O'Donnell '369 in the manner described in the Office Action

The Office Action argues that Conte teaches that it would be obvious to move

O'Donnell's techniques from software to hardware. However, O'Donnell's software techniques

cannot practically be implemented in hardware.

First, O'Donnell's software requires frequent modification that is not practical in

hardware. O'Donnell '369 teaches that his software "observer" is specifically "generated" and

tailored to each profiling configuration, on an as-needed basis. Col. 7 lines 32-38 (after a user

specifies the "particulars" of what is to be observed, O'Donnell '369 "generates the observer");

col. 8 lines 52-67 (O'Donnell's "finite state automaton" is a substitute executable for specific

portions of a program being profiled); see also col. 7 lines 2-3; and col. 6 lines 64 to col. 7 line 8.

Hardware cannot be generated on such a "throw-away" basis in any cost-effective manner, and typically requires several weeks of turn-around time. Hardware structures in this field are not generated with respect to "particulars" of one user's one-time need; rather, hardware is designed to be general and re-useable. Accordingly, Applicants' specification discloses a few simple, general structures and techniques that can be adapted to profile many different programs. See, e.g., Fig. 5a and control logic at Fig. 5b (neither of which are specific to any program to be profiled); see also Figs. 4g, 4h, and 4i, and the associated portions of the specification, at pages 73-89.

A second key difference between hardware and software is cost. The cost of software goes up roughly linearly with capability. In contrast, the cost of hardware goes up roughly as the cube of die size, or roughly as the cube of chip capability. See lecture slides from the University of California at Berkeley, available at vco.ett.utu.fi/courses/ ETT_2015/kalvot/luento4.pdf, and attached hereto as Exhibit A. Therefore, most CPU chip designs avoid large on-chip structures that are not directly related to the core tasks of executing instructions. Profiling is a non-core function, and thus CPU designers avoid any implementation that requires large on-chip structures.

Because O'Donnell '369 uses software, O'Donnell can use large structures that are not amenable to hardware implementation. For a first example, O'Donnell's entire program and the finite state automaton are linked together for profiling. Col. 7 lines 43-45. Implementing an entire program in hardware would not be recognized as desirable by one of ordinary skill. For a second example, O'Donnell's "finite state automaton" executes instead of, and is therefore at least as complex as, the portion of the program being profiled, col. 8 lines 52-67, and O'Donnell's structure includes "data breakpoints" and "instruction breakpoints." Col. 7 lines 51-54. Because O'Donnell '369 uses an entirely-software approach, he has no motivation to teach techniques that could practically be implemented in hardware. Implementing O'Donnell's large, non-general structures in hardware would not be practical, let alone desirable, to one of ordinary skill.

In contrast, Applicant's disclosure teaches several design techniques for reducing the amount of hardware required for hardware profiling. See, e.g., the simple state machine of Fig.

5a, and the simple logic structure and reasonably small registers of Fig. 5b.[1] These techniques make the invention of claim 1 practical.

No adaptation of O'Donnell's techniques to hardware would be desirable, or could "reasonably be expected to succeed," MPEP §§ 2143.01, 2143.02[2]. Therefore, no such modification of O'Donnell '369 would be obvious, and no obviousness rejection based on O'Donnell '369 is warranted.

B.   Neither O'Donnell '369 nor Conte discloses "detecting and recording occurring under control of hardware of the computer without compiler assistance ... or event-by-event software intervention"

Applicant was first to recognize that a profiler could be implemented in hardware "without compiler assistance ... or event-by-event software intervention" and to conceive a practical implementation. In contrast, both O'Donnell '369 and Conte rely on software (either a compiler or other program translator, a run-time system, or both) for event-by-event collection of profile information.

As the Office Action itself acknowledges, O'Donnell '369 does not disclose profiling in which the "detecting and recording [occurs] under control of hardware," without the assistance of a compiler or event-by-event software intervention.[3] Instead, O'Donnell teaches that his profiling is performed in an "observer," which is entirely implemented in software. Abstract; col. 8 lines 50-67; see also col. 7 lines 34-43; col. 8 lines 1-51; col. 9 lines 39-40. O'Donnell teaches that his software performs the event-by-event "detecting and recording." *Id.*

Likewise, Conte describes an implementation that relies on fairly extensive special-purpose software, including modifications that must be made to a compiler. Conte describes the

---

[1] The claims are not limited to the implementations disclosed. This observation is merely offered to show the concerns that confront one of ordinary skill, and to show that such a one would not be motivated to translate the large tables and other structures of O'Donnell '369 to hardware.

[2] The Office Action omits any consideration of "reasonable expectation of success." Such a showing is required by MPEP §§ 2143, 2143.02. Without a showing of "reasonable expectation of success," the written Office Action is incomplete, and no rejection under § 103(a) exists.

[3] The claim does not exclude all software assistance, only assistance from a compiler, and event-by-event software intervention. *See, e.g.*, page 75 lines 14-15.

support that a compiler must put into object code compiled for profiling, at page 15, col. 1.

Starting at the top of the column, Conte states:

> The specific procedure for producing a WCFG is as follows: (1) A program is compiled with a special identifier token (magic number), indicating it contains a table of CFG arcs.

This "table of CFG arcs" is generated by the compiler to support in profiling. Conte § 2.2 (pages 13-14). Conte's caption for his § 3.1 states that a compiler must "adjust" code "to support arc-based profiling." Conte's § 3.1 describes how "switch" statements in C must be specially compiled in order to use his profiling. Conte's § 3.4 describes how a compiler must be tuned to avoid "code explosion." ("Traces," "trace selection," "code explosion," and "grouping blocks," discussed in §§ 3-3.4, are terms that apply to compilers, not to hardware. See Conte's §§ 1 and 2, and the references cited in Conte's bibliography.)

The embodiment of claim 1 disclosed in Applicant's specification provides capabilities that could not be realized by any combination of O'Donnell '369 and Conte. For example, Applicant's specification describes how a program may be profiled, even with no access to the source code. *E.g.*, at page 61 lines 1-3 ("Because the X86 instruction text is typically an off-the-shelf commercial binary, profiler **400** operates without modifying the X86 binary, or recompiling source code into special-purpose profileable X86 instruction text."). In contrast, both Conte and O'Donnell rely on special processing of the source code. Conte at page 15, col. 1, lines 2-3; O'Donnell '369 at col. 6, lines 41-42 and 64-66.

Neither O'Donnell '369 nor Conte teach hardware that can profile "without compiler assistance ... or event-by-event software intervention" as recited in claim 1. Thus, the combination does not render claim 1 obvious.

Claim 24 recites similar language, and is patentable for similar reasons.

IV.   Dependent claims

Applicant renews the traverse of the rejection of the dependent claims raised at pages 3-4 of the Response of April 2003. The written Office Action does not set out a correspondence between elements of the claim and the references. The relationship between the claims and the indicated portions of the references is not apparent. Applicant cannot determine the grounds of

rejection, let alone respond. Applicant requests that the Examiner respond to all material traversed. MPEP § 707.07(f).

Dependent claims 2-23 and 25-45 are patentable with the independent claims discussed above. In addition, the dependent claims recite additional features that further distinguish the art.

In view of the amendments and remarks, Applicant respectfully submits that the claims are in condition for allowance. Applicant requests that the application be passed to issue in due course. The Examiner is urged to telephone Applicant's undersigned counsel at the number noted below if it will advance the prosecution of this application, or with any suggestion to resolve any condition that would impede allowance. Enclosed is Petition to ensure timeliness. In the event that further extension of time is required, Applicant petitions for that extension of time required to make this response timely. Kindly charge any additional fee, or credit any surplus, to Deposit Account No. 23-2405, Order No. 114596-10-4017.

Respectfully submitted,

WILLKIE FARR & GALLAGHER, LLP

Dated: <u>March 10, 2004</u>

By: _____
David E. Boundy
Registration No. 36,461

WILLKIE FARR & GALLAGHER, LLP
787 Seventh Ave.
New York, New York 10019
(212) 728-8000
(212) 728-8111 Fax